The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1.   A method for sending a message via a high-performance message queue, comprising:

providing a message queue associated with a context;

executing a user interface thread associated with said context;

receiving a request from said user interface thread to send a message to a second user interface thread;

determining whether said second user interface thread is associated with said context; and

in response to determining that said second user interface thread is associated with said context, sending said message from said user interface thread directly to said second user interface thread, thereby bypassing said message queue.

2.   The method of Claim 1, further comprising:

in response to determining that said second user interface thread is not associated with said context, atomically adding said message to a queue associated with a second context.

3.   The method of Claim 2, further comprising:

atomically providing an indication to said second context that a message has been added to said queue associated with said second context.

4.   The method of Claim 3, further comprising:

waiting for an indication that said message added to said queue associated with said second context has been processed; and

processing additional messages while waiting for said indication.

5.   The method of Claim 4, wherein atomically adding said message to a queue associated with a second context comprises locking said message and atomically adding said message to a singly-linked list associated with said context.

6.   A computer-controlled apparatus capable of performing the method of any one of Claims 1-5.

7.   A computer-readable medium comprising instructions which, when executed by a computer, cause the computer to perform the method of any one of Claims 1-5.

8.   A method for posting a message via a high-performance message queue, comprising:

providing a message queue associated with a context;

executing a user interface thread associated with said context;

receiving a request from said user interface thread to post a message to a queue associated with a second context;

atomically adding said message to said queue associated with said second context; and

atomically providing an indication to said second context that a message has been posted to said queue associated with said second context.

9.   The method of Claim 8, further comprising:

validating parameters associated with said message;

determining a processing function to handle the dequeuing of said message; and

completing a message entry for said message including said validated parameters and the identity of said processing function.

10.   The method of Claim 9, wherein atomically adding said message to said queue associated with said second context comprises locking said message entry and atomically adding said message entry to a singly-linked list associated with said context.

11.   A computer-controlled apparatus capable of performing the method of any one of Claims 8-10.

12.   A computer-readable medium comprising instructions which, when executed by a computer, cause the computer to perform the method of any one of Claims 8-10.

13.   A method for processing the contents of a high-performance message queue, comprising:

receiving an indication that the contents of a post message queue and a sent message queue should be processed by a user interface thread;

atomically determining whether a thread other than said user interface thread is currently processing said post message queue or said sent message queue;

in response to determining that no other thread is processing said post or sent message queues, indicating that said user interface thread is processing said post and sent message queues, atomically processing said sent message queue and said post message queue; and

when processing of said post and sent message queues has completed, providing an indication that no thread is processing said post or said sent message queues.

14.   The method of Claim 13, wherein processing said post message queue comprises:

determining whether said post message queue is empty;

in response to determining that said post message queue is not empty, atomically extracting a singly linked list that comprises said post message queue;

reversing said singly linked list; and

extracting one or more message entries contained in said singly linked list.

15. The method of Claim 13, wherein processing said sent message queue comprises:

determining whether said sent message queue is empty;

in response to determining that said sent message queue is not empty, atomically extracting a singly linked list that comprises said sent message queue;

reversing said singly linked list; and

extracting one or more message entries contained in said singly linked list.

16. A computer-controlled apparatus capable of performing the method of any one of Claims 13-15.

17. A computer-readable medium comprising instructions which, when executed by a computer, cause the computer to perform the method of any one of Claims 13-15.

18. A method for providing a queue bridge between a window manager supporting high-performance queues and a legacy window manager, comprising:

determining whether a message is ready for said window manager supporting high-performance message queues;

in response to determining that a message is not ready for said window manager supporting high-performance message queues, determining whether a message is ready for said legacy window manager; and

in response to determining that a message is not ready for said legacy window manager, performing idle-time processing and determining whether a message is ready for either said high-performance window manager or said legacy window manager.

19. The method of Claim 18, further comprising:

in response to determining that a message is ready for said legacy window manager, processing the next available legacy window manager message.

20. The method of Claim 19, further comprising:

in response to determining that a message is ready for said window manager supporting high-performance queues, extracting and processing all messages available for said window manager supporting high-performance queues.

21. A computer-controlled apparatus capable of performing the method of any one of Claims 18-20.

22. A computer-readable medium comprising instructions which, when executed by a computer, cause the computer to perform the method of any one of Claims 18-20.